Notice of Allowability	Application No.	Applicant(s)	
	10/738,377	AUERBACH, JOSHUA S.	
	Examiner	Art Unit	
	Michael D. Pham	2167	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to *10/25/06*.

2. ☒ The allowed claim(s) is/are *1-8, 10-12, 14-19, 21-23*.

3. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a) ☐ All b) ☐ Some* c) ☐ None of the:
      1. ☐ Certified copies of the priority documents have been received.
      2. ☐ Certified copies of the priority documents have been received in Application No. _____ .
      3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

4. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.

5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
      1) ☐ hereto or 2) ☐ to Paper No./Mail Date _____.
   (b) ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____.

   Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).

6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☐ Notice of References Cited (PTO-892)
2. ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
3. ☐ Information Disclosure Statements (PTO/SB/08), Paper No./Mail Date _____
4. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
5. ☐ Notice of Informal Patent Application
6. ☒ Interview Summary (PTO-413), Paper No./Mail Date _____ .
7. ☒ Examiner's Amendment/Comment
8. ☒ Examiner's Statement of Reasons for Allowance
9. ☐ Other _____.

DETAILED ACTION

1. Claims 1-8, 10-12, 14-19, 21-23 are pending in this office action.

EXAMINER'S AMENDMENT

2. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephone interview with Mr. Wendell Peete (reg. 52,108) on January 3, 2007.

In the claims:

Please replace original claims 1, 4, 5, 6, 7, 8, 12, 14, 16, 17, 18, 19, and 23 with amended claims 1, 4, 5, 6, 7, 8, 12, 14, 16, 17, 18, 19, and 23.

Claim 1:

A method comprising organizing a byte stream of an information structure, said information structure having a schema and an in-memory representation, said schema having a schema tree representation with a plurality of schema nodes, said schema nodes including at least one leaf and at least one interior node, the step of organizing comprising the steps of:

computing a layout from the schema tree representation depth-first enumeration of leaf nodes of the schema;

serializing the byte stream from the in-memory representation while grouping together all scalar items from the in-memory representation corresponding to each schema node, wherein the step of serializing the byte stream further comprises the steps of:

retrieving a location in the byte stream for an element of the in-memory representation corresponding to a first schema leaf node in depth first order from the layout;

converting the element to bytes in the byte stream according to a number of elements corresponding to the schema leaf node, storing a result during said converting the element; and

accessing information from the byte stream by using the layout and offset calculations, wherein the step of accessing information further comprises the steps of:

scanning a list of key values representing a table column serialized within the byte stream to determine an index position; and

using the index position in conjunction with offset calculations and offset tables serialized at the start of lists within the byte stream to find information in lists representing non-key table columns.

Claim 4:

The method as recited in claim 1, wherein the step of computing [[a]] the layout comprises:

establishing a fixed length portion of the byte stream, the fixed length portion having a slot for each enumerated schema leaf node having a predecessor in depth-first numbering requiring varying length encoding; and

establishing a varying length portion of the byte stream following the fixed length portion, the varying length portion having successive areas for said each enumerated schema leaf node.

Claim 5:

The method as recited in claim 1 wherein the interior nodes of said schema tree representation are restricted to list and tuple nodes, and the leaf nodes comprise scalar types and dynamic types.

Claim 6:

The method as recited in claim 1, wherein the step of serializing the byte stream comprises:

determining a correspondence between the in-memory representation and the schema tree representation;

initializing the byte stream by reserving a fixed length portion and pointing to a beginning of a variable length portion;

[[retrieving a location in the byte stream for an element of the in-memory representation information corresponding to a first schema leaf node in depth first order from the layout;]]

[[converting the element to bytes in the byte stream according to a number of elements corresponding to the schema leaf node;]]

and

repeating the steps of retrieving and converting for all schema leaf nodes in depth-first order.

Claim 7:

The method as recited in claim [[6]] 1, wherein the step of converting elements to bytes comprises recording a nested list of tuples in column order rather than row order, resulting in a set of nested lists.

Claim 8:

The method as recited in claim [[6]] 1, wherein the step of converting elements to bytes comprises preceding each list of varying length items with an offset table allowing any element of said each list to be reached in constant time from a head of said each list.

Claim 12:

[[A computer program product that includes a computer readable medium useable by a processor, the medium having stored thereon a sequence of instructions which, when executed by the processor, causes the processor to organize a byte stream of an information structure, wherein the computer program product executes the steps of:]]

A computer program product stored in a computer readable storage medium having stored thereon a sequence of instructions which, when executed by a processor, causes the processor to organize a byte stream of an information structure, wherein the computer program product executes the steps of:

computing a layout from a [[the]] schema tree representation by depth-first enumeration of leaf nodes of the schema

serializing the byte stream from a [[the]] in-memory representation while grouping together all scalar items from the in-memory representation corresponding to each schema node, wherein the step of serializing the byte stream further comprises the steps of:

retrieving a location in the byte stream for an element of the in-memory representation corresponding to a first schema leaf node in depth first order from the layout;

converting the element to bytes in the byte stream according to a number of elements corresponding to the schema leaf node, storing a result during said converting the element; and

accessing information from the byte stream by using the layout and offset calculations, wherein the step of accessing information further comprises the steps of:

scanning a list of key values representing a table column serialized within the byte stream to determine an index position; and

using the index position in conjunction with offset calculations and offset tables serialized at the start of lists within the byte stream to find information in lists representing non-key table columns.

Claim 14:

An apparatus comprising a serializer/deseralizer for a byte stream form of an information structure, said information structure having a schema and an in-memory representation, said schema having a schema tree representation with a plurality of schema nodes, said schema nodes including at least one leaf and at least one interior node, the serializer/deserializer comprising:

a processor for computing a layout from the schema tree representation by depth-first enumeration of leaf nodes of the schema;

a serializer for serializing the byte stream from the in-memory representation while grouping together all scalar items from the in-memory representation corresponding to each schema node, wherein the serializer further comprises a lookup module to retrieve a location in the byte stream for an element of the in-memory representation corresponding to a first schema leaf node in depth first order from the layout;

a converter to convert the element to bytes in the byte stream according to a number of elements corresponding to the schema leaf node, wherein all schema leaf nodes are retrieved and converted in depth-first order, storing a result during said converting the element; and

a selective de-serializer for accessing information from the byte stream by using the layout and offset calculations, wherein the selective de-serializer scans a list of key values representing a table column serialized within the byte stream to determine an index position, and uses the index position in conjunction with offset calculations and offset tables serialized at the [[starts]] start of lists within the byte stream to find information in lists representing non-key table columns.

Claim 16:

The apparatus as recited in claim 14, wherein the processor comprises:

a module for establishing a fixed length portion of the byte stream, the fixed length portion having a slot for each enumerated schema leaf node having a predecessor in depth-first numbering requiring varying length encoding; and

for establishing a varying length portion of the byte stream following the fixed length portion, the varying length portion having successive areas for said each enumerated schema leaf node.

Claim 17:

The apparatus as recited in claim 14, wherein the serializer comprises:

a reconciling module to determine a correspondence between the in-memory representation and the schema tree representation;

an initialization module to initialize the byte stream by reserving a fixed length portion and pointing to a beginning of a variable length portion[[;]]

[[a lookup module to retrieve a location in the byte stream for an element of the in-memory representation information corresponding to a first schema leaf node in depth first order from the layout;]]

[[a converter to convert the element to bytes in the byte stream according to a number of elements corresponding to the schema leaf node, wherein all schema leaf nodes are retrieved and converted in depth-first order]].

Claim 18:

The apparatus as recited in claim [[17]] 14, wherein the converter comprises a recorder to record a nested list of tuples in column order rather than row order, resulting in a set of nested lists.

Claim 19:

The apparatus as recited in claim [[17]] 14, wherein the converter precedes each list of varying length items with an offset table allowing any element of said each list to be reached in constant time from a head of said each list.

Claim 23:

[[A computer program product that includes a computer readable medium useable by a processor, the medium having stored thereon a sequence of instructions which, when executed by the processor, causes the processor to organize a byte stream form of an information structure, wherein the computer program product executes the steps of:]]

A computer program product stored in a computer readable storage medium having stored thereon a sequence of instructions which, when executed by a processor, causes the processor to organize a byte stream form of an information structure, wherein the computer program product executes the steps of:

computing a layout from a [[the]] schema tree representation by depth-first enumeration of leaf nodes of the schema;

serializing the byte stream from a [[the]] in-memory representation while grouping together all scalar items from the in-memory representation corresponding to each schema node, wherein the step of serializing the byte stream further comprises the steps of:

retrieving a location in the byte stream for an element of the in-memory representation corresponding to a first schema leaf node in depth first order from the layout;

converting the element to bytes in the byte stream according to a number of elements corresponding to the schema leaf node, storing a result during said converting the element;

and

accessing information from the byte stream by using the layout and offset calculations,

wherein a selective de-serializer scans a list of key values representing a table column serialized within the byte stream to determine an index position, and

using the index position in conjunction with offset calculations and offset tables serialized at the [[starts]] start of lists within the byte stream to find information in lists representing non-key table columns.

Allowable Subject Matter

3. Claims 1-8, 10-12, 14-19, 21-23 are allowed.

The prior art of record, alone or in combination, does not teach or fairly suggest the combination of steps recited in independent claims 1 and 12 wherein "serializing the byte stream from the in-memory representation while grouping together all scalar items from the in-memory representation to each schema node, wherein the step of serializing the byte stream further comprises the steps of:retrieving a location in the byte stream for an element of the in-memory representation information corresponding to a first schema leaf node in depth first order from the layout; converting the element to bytes in the byte stream according to a number of elements corresponding to the schema leave node, storing a result during said converting the element; accessing information from the byte stream by using the layout and offset calculations wherein

the step of accessing information further comprises the steps of: scanning a list of key values representing a table column serialized within the byte stream to determine an index position; and using the index position in conjunction with offset calculations and offset tables serialized at the start of lists within the byte stream to find information in lists representing non-key table columns. " in combination with all other claimed elements in claims 1 and 12.

The prior art of record, alone or in combination, does not fairly suggest the combination of steps recited in independent claims 14 and 23 wherein "a serializer for serializing the byte stream from the in-memory representation while grouping together all scalar items from the in-memory representation corresponding to each schema node, wherein the serializer further comprises a lookup module to retrieve a location in the byte stream for an element of the in-memory representation information corresponding to a first schema leaf node in depth first order from the layout; a converter to convert the element to bytes in the byte stream according to a number of elements corresponding to the schema leaf node, wherein all schema leaf nodes are retrieved and converted in depth-first order, storing a result during said converting the element; a selective de-serializer for accessing information from the byte stream by using the layout and offset calculations, wherein the selective de-serializer scans a list of key values representing a table column serialized within the byte stream to determine an index position, and uses the index position in conjunction with offset calculations and offset tables serialized at the start of lists within the byte stream to find information in lists representing non-key table columns" in combination with all other claimed elements in claims 14 and 23.

The dependent claims, bring definite, further limiting, and fully enabled by the specification are also allowed.

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

Contact Information

4. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Michael D. Pham whose telephone number is (571)272-3924. The examiner can normally be reached on Monday - Friday 9am - 5:00pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Cottingham can be reached on 571-272-7079. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Michael Pham
Art Unit 2167
Examiner MP.

Cam Y. Truong
Art Unit 2162
Primary Examiner

CT.

John Cottingham
Art Unit 2167
Supervisor

JOHN COTTINGHAM
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100